International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org Volume 3 Issue 4 || April 2014 || PP.01-10

Review on Denoising techniques for the AWGN signal introduced in a stationary image

Apeksha Jain[1], Mr. S. G. Kerhalkar[2], Mr. Mohammed Ahmed [3]
[1,] M .Tech scholar [2,] Head of Department [3,] Assistant Professor
[1, 2, 3,] Department of Electronics & Communication Engineering, Oriental Institute of science & Technology, Bhopal

ABSTRACT:*Visual information transmitted in the form of digital images is becoming a major method of communication in the modern age, but the image obtained after transmission is often corrupted with noise. The received image needs processing before it can be used in applications. Image denoising involves the manipulation of the image data to produce a visually high quality image. This paper reviews the existing denoising algorithms, such as filtering approach; wavelet based approach and performs their comparative study. Different noise models including additive and multiplicative types are used. They include Gaussian noise, salt and pepper noise, speckle noise and Brownian noise. Selection of the denoising algorithm is application dependent. Hence, it is necessary to have knowledge about the noise present in the image so as to select the appropriate denoising algorithm. The filtering approach seems to be a better choice when the image is corrupted with salt and pepper noise. The wavelet based approach finds applications in denoising images corrupted with Gaussian noise. In this review paper denoising techniques for the AWGN signal introduced in an image have been studied.*

KEYWORDS :*AWGN, Denoising, Blurring, Noise, Filtering, DWT, threshold*.

I. INTRODUCTION

A very large portion of digital image processing is devoted to image restoration. This includes research in algorithm development and routine goal oriented image processing. Image restoration is the removal or reduction of degradations that are incurred while the image is being obtained. Degradation comes from blurring as well as noise due to electronic and photometric sources.[1][2] Blurring is a form of bandwidth reduction of the image caused by the imperfect image formation process such as relative motion between the camera and the original scene or by an optical system that is out of focus. When aerial photographs are produced for remote sensing purposes, blurs are introduced by atmospheric turbulence, aberrations in the optical system and relative motion between camera and ground. In addition to these blurring effects, the recorded image is corrupted by noises too. A noise is introduced in the transmission medium due to a noisy channel, errors during the measurement process and during quantization of the data for digital storage. Each element in the imaging chain such as lenses, film, digitizer, etc. contributes to the degradation [3].

Image denoising is often used in the field of photography or publishing where an image was somehow degraded but needed to be improved before it can be printed. For this type of application we need to know something about the degradation process in order to develop a model for it. When we have a model for the degradation process, the inverse process can be applied to the image to restore it back to the original form. This type of image restoration is often used in space exploration to help eliminate artifacts generated by mechanical jitter in a spacecraft or to compensate for distortion in the optical system of a telescope [4]. Image denoising finds applications in fields such as astronomy where the resolution limitations are severe, in medical imaging where the physical requirements for high quality imaging are needed for analyzing images of unique events, and in forensic science where potentially useful photographic evidence is sometimes of extremely bad quality.

Let us now consider the representation of a digital image. A 2-dimensional digital image can be represented as a 2-dimensional array of data $s(x,y)$, where (x,y) represent the pixel location. The pixel value corresponds to the brightness of the image at location (x,y). Some of the most frequently used image types are binary, gray-scale and color.Binary images are the simplest type of images and can take only two discrete values, black and white. Black is represented with the value '0' while white with '1'. Note that a binary image is generally created from a gray-scale image. A binary image finds applications in computer vision areas where the general shape or outline information of the image is needed. They are also referred to as 1 bit/pixel images [5].Gray-scale images are known as monochrome or one-color images. The images used for experimentation purposes in this thesis are all gray-scale images. They contain no color information. They represent the brightness of the image. This image contains 8 bits/pixel data, which means it can have up to 256 (0-255) different brightness levels.

A '0' represents black and '255' denotes white. In between values from 1 to 254 represent the different gray levels. As they contain the intensity information, they are also referred to as intensity images.Color images are considered as three band monochrome images, where each band is of a different color. Each band provides the brightness information of the corresponding spectral band. Typical color images are red, green and blue images and are also referred to as RGB images. This is a 24 bits/pixel image.
The main aim of this paper is to review all the existing methodology which are used for estimation of the uncorrupted image from the distorted or noisy image, and is also referred to as image "denoising". There are various methods to help restore an image from noisy distortions. Selecting the appropriate method plays a major role in getting the desired image. The denoising methods tend to be problem specific. For example, a method that is used to denoise satellite images may not be suitable for denoising medical images. In this paper it is proposed that a study would made on the various denoising algorithms In case of image denoising methods, the characteristics of the degrading system and the noises are assumed to be known.

II. BASIC NOISE THEORY

Noise is defined as an unwanted signal that interferes with the communication or measurement of another signal. A noise itself is an information-bearing signal that conveys information regarding the sources of the noise and the environment in which it propagates.

There are many types and sources of noise or distortions and they include [7][8]:

- **Electronic noise** such as thermal noise and shot noise.
- **Acoustic noise** emanating from moving, vibrating or colliding sources such as revolving machines, moving vehicles, keyboard clicks, wind and rain.
- **Electromagnetic noise** that can interfere with the transmission and reception of voice, image and data over the radio-frequency spectrum.
- **Electrostatic noise generated** by the presence of a voltage.
- **Quantization noise** and lost data packets due to network congestion.

Signal distortion is the term often used to describe a systematic undesirable change in a signal and refers to changes in a signal from the non-ideal characteristics of the communication channel, signal fading reverberations, echo, and multipath reflections and missing samples. Depending on its frequency, spectrum or time characteristics, a noise process is further classified into several categories:

II.I Additive and Multiplicative Noises

Noise is present in an image either in an additive or multiplicative form. An additive noise follows the rule [9]:

$$w(x,y) = s(x,y) + n(x,y)$$

While the multiplicative noise satisfies

$$w(x,y) = s(x,y) X\, n(x,y)$$

Where s(x,y) is the original signal, n(x,y) denotes the noise introduced into the signal to produce the corrupted image w(x,y), and (x,y) represents the pixel location.

II.II Gaussian Noise

Gaussian noise is evenly distributed over the signal this means that each pixel in the noisy image is the sum of the true pixel value and a random Gaussian distributed noise value. As the name indicates, this type of noise has a Gaussian distribution, which has a bell shaped probability distribution function given by,

$$F(g) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(g-m)^2}{2\sigma^2}}$$

Where g represents the gray level, m is the mean or average of the function and σ is the standard deviation of the noise. Graphically, it is represented as shown in Figure 1.

When introduced into an image, Gaussian noise with zero mean and variance as 0.05 would look as in Figure 1 [3].

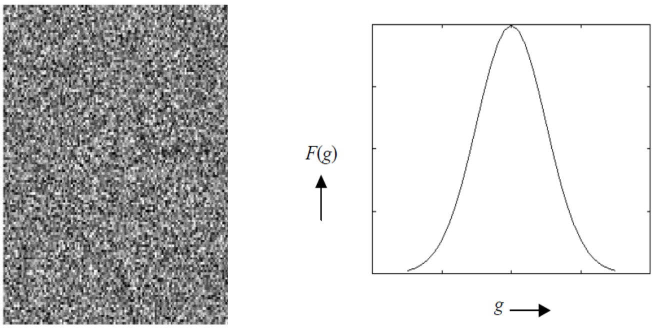

Figure 1: (a) Gaussian noise (mean=0, variance 0.05), (b) Gaussian distribution

II.III Salt and Pepper Noise

Salt and pepper noise is an impulse type of noise, which is also referred to as intensity spikes. This is caused generally due to errors in data transmission. It has only two possible values, *a* and *b*. The probability of each is typically less than 0.1. The corrupted pixels are set alternatively to the minimum or to the maximum value, giving the image a “salt and pepper” like appearance. Unaffected pixels remain unchanged. For an 8-bit image, the typical value for pepper noise is 0 and for salt noise 255. The salt and pepper noise is generally caused by malfunctioning of pixel elements in the camera sensors, faulty memory locations, or timing errors in the digitization process. The probability density function for this type of noise is shown in Figure 2.

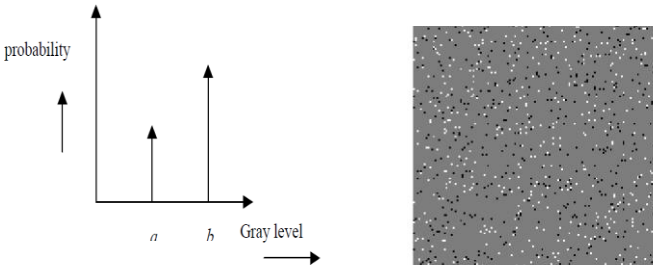

Figure 2: (a) Salt and pepper noise (b) PDF for salt and pepper noise

II.IV Speckle Noise

Speckle noise is a multiplicative noise. This type of noise occurs in almost all coherent imaging systems such as laser, acoustics and SAR (Synthetic Aperture Radar) imagery. The source of this noise is attributed to random interference between the coherent returns. Fully developed speckle noise has the characteristic of multiplicative noise. Speckle noise follows a gamma distribution and is given as:

$$F(g) = \frac{g^{\alpha-1}}{(\alpha-1)!\,\alpha^{\alpha}} e^{\frac{g}{\alpha}}$$

Where variance is $a^2\alpha$ and *g* is the gray level.
On an image, speckle noise (with variance 0.05) looks as shown in Figure 3

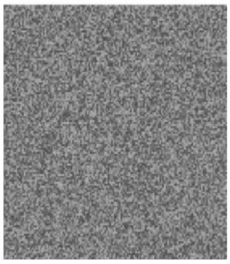

Fig. 3: Speckle noise

The gamma distribution is given below in Figure 4

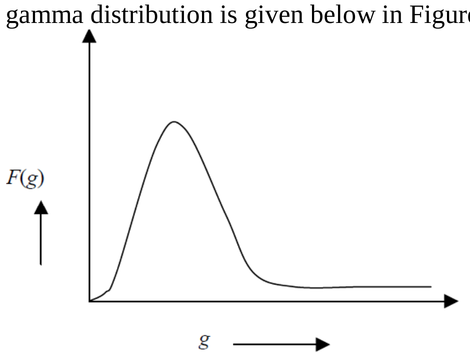

Fig. 4: Gamma distribution

II.V Brownian Noise

Brownian noise comes under the category of fractal or 1/*f* noises. The mathematical model for 1/*f* noise is fractional Brownian motion. Fractal Brownian motion is a non-stationary stochastic process that follows a normal distribution. Brownian noise is a special case of 1/*f* noise. It is obtained by integrating white noise. It can be graphically represented as shown in Figure 5. On an image, Brownian noise would look like Image 6.

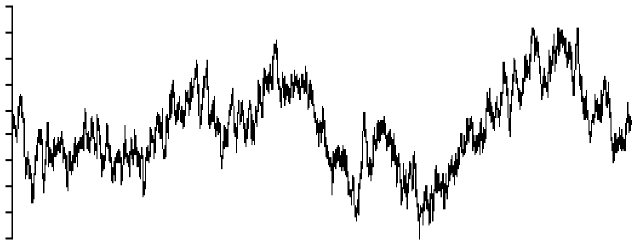

Fig. 5: Brownian noise distribution

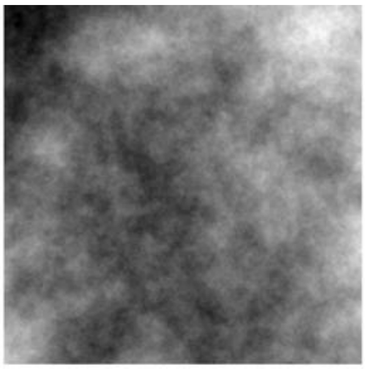

Fig. 6: Brownian noise

III. DENOISING APPROACH

III.I Linear and Nonlinear Filtering Approach

Filters play a major role in the image restoration process. The basic concept behind image restoration using linear filters is digital convolution and moving window principle. Let *w*(*x*) be the input signal subjected to filtering, and *z*(*x*) be the filtered output. If the filter satisfies certain conditions such as linearity and shift invariance, then the output filter can be expressed mathematically in simple form as

$$Z(X) = \int W(t) h(x-t) dt$$

Where *h*(*t*) is called the point spread function or impulse response and is a function that completely characterizes the filter. The integral represents a convolution integral and, in short, can be expressed as

$$z = w * h.$$

For a discrete case, the integral turns into a summation as

$$z(i) = \sum_{-\infty}^{+\infty} w(t)h(i-t)$$

III.II Mean Filter

A mean filter acts on an image by smoothing it; that is, it reduces the intensity variation between adjacent pixels. The mean filter is nothing but a simple sliding window spatial filter that replaces the center value in the window with the average of all the neighboring pixel values including itself. By doing this, it replaces pixels that are unrepresentative of their surroundings. It is implemented with a convolution mask, which provides a result that is a weighted sum of the values of a pixel and its neighbors. It is also called a linear filter. The mask or kernel is a square. Often a 3× 3 square kernel is used. If the coefficients of the mask sum up to one, then the average brightness of the image is not changed. If the coefficients sum to zero, the average brightness is lost, and it returns a dark image. The mean or average filter works on the shift-multiply-sum principle this principle in the two-dimensional image can be represented as shown below:

h_1	h_2	h_3
h_4	h_5	h_6
h_7	h_8	h_9

Fig. 7: Filter mask

Note that the coefficients of this mask sum to one, so the image brightness is retained, and the coefficients are all positive, so it will tend to blur the image. Mean filters are popular for their simplicity and ease of implementation. The mean filter is used in applications where the noise in certain regions of the image needs to be removed. In other words, the mean filter is useful when only a part of the image needs to be processed.

III.III LMS Adaptive Filter

An adaptive filter does a better job of denoising images compared to the averaging filter. The fundamental difference between the mean filter and the adaptive filter lies in the fact that the weight matrix varies after each iteration in the adaptive filter while it remains constant throughout the iterations in the mean filter. Adaptive filters are capable of denoising non-stationary images, i.e., images that have abrupt changes in intensity. Such filters are known for their ability in automatically tracking an unknown circumstance or when a signal is variable with little a priori knowledge about the signal to be processed. In general, an adaptive filter iteratively adjusts its parameters during scanning the image to match the image generating mechanism. This mechanism is more significant in practical images, which tend to be non-stationary.

Compared to other adaptive filters, the Least Mean Square (LMS) adaptive filter is known for its simplicity in computation and implementation. The basic model is alinear combination of a stationary low-pass image and a non-stationary high-pass component through a weighting function. Thus, the function provides compromise between resolution of genuine features and suppression of noise.

The LMS adaptive filter incorporating a local mean estimator works on the following concept. A window, *W*, of size $m \times n$ is scanned over the image. The mean of this window, μ, is subtracted from the elements in the window to get the residual matrix W_r.

$$W_r = W - \mu$$

III.IV Median Filter

A median filter belongs to the class of nonlinear filters unlike the mean filter. The median filter also follows the moving window principle similar to the mean filter. A 3× 3, 5× 5, or 7× 7 kernel of pixels is scanned over pixel matrix of the entire image. The median of the pixel values in the window is computed, and the center pixel of the window is replaced with the computed median. Median filtering is done by, first sorting all the pixel values from the surrounding neighborhood into numerical order and then replacing the pixel being considered with the middle pixel value. Note that the median value must be written to a separate array or buffer so that the results are not corrupted as the process is performed.

123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

Fig 8: Concept of median filtering
Neighborhood values:

115,119,120,123,124,125,126,127,150
Median value: 124

The central pixel value of 150 in the 3×3 window shown in Figure 8 is rather unrepresentative of the surrounding pixels and is replaced with the median value of 124. The median is more robust compared to the mean. Thus, a single very unrepresentative pixel in a neighborhood will not affect the median value significantly. Since the median value must actually be the value of one of the pixels in the neighborhood, the median filter does not create new unrealistic pixel values when the filter straddles an edge. For this reason the median filter is much better at preserving sharp edges than the mean filter. These advantages aid median filters in denoising uniform noise as well from an image.

III.V Discrete Wavelet Transform (DWT) – Principles

Wavelets are mathematical functions that analyze data according to scale or Resolution. They aid in studying a signal in different windows or at different resolutions. For instance, if the signal is viewed in a large window, gross features can be noticed, but if viewed in a small window, only small features can be noticed. Wavelets provide some advantages over Fourier transforms. For example, they do a good job in approximating signals with sharp spikes or signals having discontinuities. Wavelets can also model speech, music, video and non-stationary stochastic signals. Wavelets can be used in applications such as image compression, turbulence, human vision, radar, earthquake prediction, etc. The term "wavelets" is used to refer to a set of Ortho-normal basis functions generated by dilation and translation of scaling function φ and a mother wavelet ψ. The finite scale multi resolution representation of a discrete function can be called as a discrete wavelet transform. DWT is a fast linear operation on a data vector, whose length is an integer power of 2. This transform is invertible and orthogonal, where the inverse transform expressed as a matrix is the transpose of the transform matrix. The wavelet basis or function, unlike sine and cosines as in Fourier transform, is quite localized in space. But similar to sine and cosines, individual wavelet functions are localized in frequency. The Ortho-normal basis or wavelet basis is defined as

$$\varphi_{(j,k)}(x) = 2^{\frac{j}{2}}\varphi(2^{j}x - k)$$

The scaling function is given as

$$\emptyset_{(j,k)}(x) = 2^{\frac{j}{2}}\varphi(2^{j}x - k)$$

Where ψ is called the wavelet function and *j* and *k* are integers that scale and dilate the wavelet function. The factor '*j*' is known as the scale index, which indicates the wavelet's width. The location index *k* provides the position. The wavelet function is dilated by powers of two and is translated by the integer *k*. In terms of the wavelet coefficients, the wavelet equation is

$$\varphi(x) = \sum_{k}^{N-1} gk\sqrt{2\emptyset(2x - k)}$$

Where *g0, g1, g2*,…. are high pass wavelet coefficients. Writing the scaling equation in terms of the scaling coefficients as given below, we get

$$\emptyset(x) = \sum_{k}^{N-1} hk\sqrt{2\emptyset(2x-k)}$$

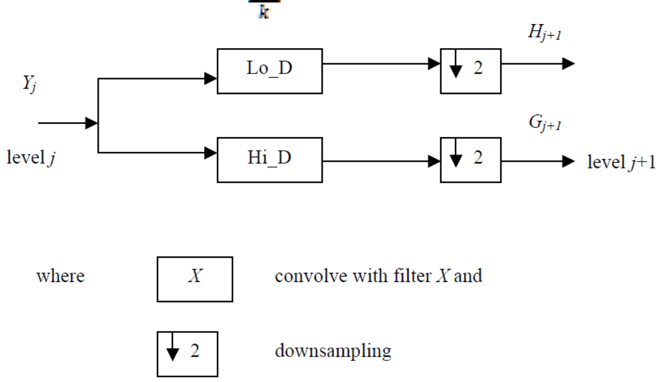

Fig. 9: A 1-Dimensional DWT - Decomposition step

DWT is the multi resolution description of an image. The decoding can be processed sequentially from a low resolution to the higher resolution. DWT splits the signal into high and low frequency parts. The high frequency part contains information about the edge components, while the low frequency part is split again into high and low frequency parts. The high frequency components are usually used for watermarking since the human eye is less sensitive to changes in edges. In two dimensional applications, for each level of decomposition, we first perform the DWT in the vertical direction, followed by the DWT in the horizontal direction. After the first level of decomposition, there are 4 sub-bands: LL1, LH1, HL1, and HH1. For each successive level of decomposition, the LL Sub-band of the previous level is used as the input. To perform second level decomposition, the DWT is applied to LL1 band which decomposes the LL1 band into the four sub- bands LL2, LH2, HL2, and HH2.

To perform third level decomposition, the DWT is applied to LL2 band which decompose this band into the four sub-bands – LL3, LH3, HL3, HH3. This results in 10 sub-bands per component. LH1, HL1, and HH1 contain the highest frequency bands present in the image tile, while LL3 contains the lowest frequency band.

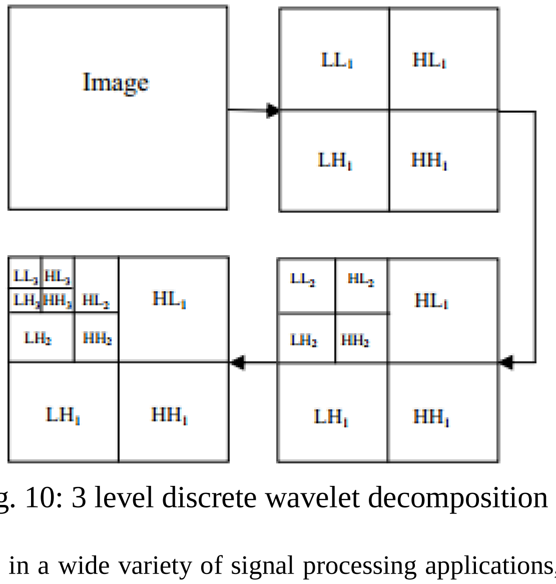

Fig. 10: 3 level discrete wavelet decomposition

DWT is currently used in a wide variety of signal processing applications, such as in audio and video compression, removal of noise in audio, and the simulation of wireless antenna distribution. Wavelets have their energy concentrated in time and are well suited for the analysis of transient, time-varying signals. Since most of the real life signals encountered are time varying in nature, the Wavelet Transform suits many applications very well.As mentioned earlier, the wavelet equation produces different wavelet families like Daubechies, Haar, Coiflets, etc. The filter lengths and the number of vanishing moments for four different wavelet families are tabulated in Table 1.

Table 1: Wavelet families and their properties

Wavelet Family	Filters length	Number of vanishing moments, *N*
Haar	2	1
Daubechies *M*	2*M*	*M*
Coiflets *M*	6*M*	2*M*-1
Symlets	2*M*	*M*

IV. WAVELET THRESHOLDING

The term wavelet thresholding is explained as decomposition of the data or the image into wavelet coefficients, comparing the detail coefficients with a given threshold value, and shrinking these coefficients close to zero to take away the effect of noise in the data. The image is reconstructed from the modified coefficients. This process is also known as the inverse discrete wavelet transform. During thresholding, a wavelet coefficient is compared with a given threshold and is set to zero if its magnitude is less than the threshold; otherwise, it is retained or modified depending on the threshold rule. Thresholding distinguishes between the coefficients due to noise and the ones consisting of important signal information.

The choice of a threshold is an important point of interest. It plays a major role in the removal of noise in images because denoising most frequently produces smoothed images, reducing the sharpness of the image. Care should be taken so as to preserve the edges of the denoised image. There exist various methods for wavelet thresholding, which rely on the choice of a threshold value. Some typically used methods for image noise removal include VisuShrink, SureShrink and BayesShrink.Prior to the discussion of these methods, it is necessary to know about the two general categories of thresholding. They are hard- thresholding and soft-thresholding types. The hard-thresholding TH can be defined as:

$$T_H = \begin{cases} X \text{ for } |x| \geq t \\ 0 \text{ in all other regions} \end{cases}$$

Here *t*is the threshold value. A plot of *TH* is shown in Figure 11:

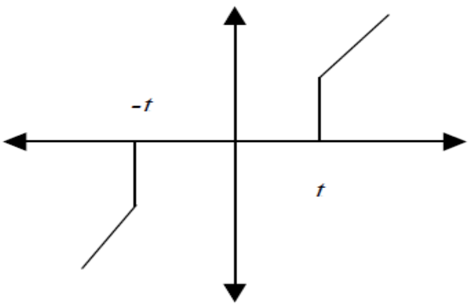

Fig. 11: Hard thresholding

Thus, all coefficients whose magnitude is greater than the selected threshold value *t*remain as they are and the others with magnitudes smaller than *t* are set to zero. It creates a region around zero where the coefficients are considered negligible. Soft thresholding is where the coefficients with greater than the threshold are shrunk towards zero after comparing them to a threshold value. It is defined as follows:

$$T_s = \begin{cases} \text{sign}(x)(|x| - t) \text{for } |x| > t \\ 0 \text{ in all other region} \end{cases}$$

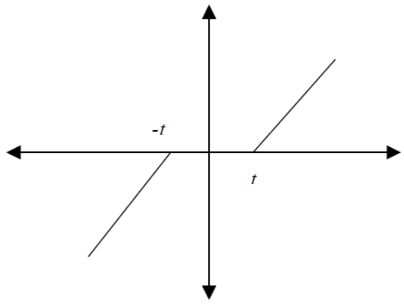

Figure 12: Soft thresholding

In practice, it can be seen that the soft method is much better and yields more visually pleasant images. This is because the hard method is discontinuous and yields abrupt artifacts in the recovered images. Also, the soft method yields a smaller minimum mean squared error compared to hard form of thresholding.

IV.I SureShrink

A threshold chooser based on Stein's Unbiased Risk Estimator (SURE) was proposed by Donoho and Johnstone and is called as SureShrink. It is a combination of the universal threshold and the SURE threshold. This method specifies a threshold value $t_j$for each resolution level j in the wavelet transform which is referred to as level dependent thresholding. The goal of SureShrink is to minimize the mean squared error, defined as:

$$MSE = \frac{1}{n^2}\sum_{x,y=1}^{n}(z(x,y) - s(x,y))^2$$

Where $z(x,y)$ is the estimate of the signal while $s(x,y)$ is the original signal without noise and n is the size of the signal. SureShrink suppresses noise by thresholding the empirical Wavelet coefficients. The SureShrink threshold t^* is defined as

$$t^* = \min\left(t, \sigma\sqrt{2\log n}\right)$$

Where t denotes the value that minimizes Stein's Unbiased Risk Estimator, σ is the noise variance and n is the size of the image. SureShrink follows the soft thresholding rule. The thresholding employed here is adaptive, i.e., a threshold level is assigned to each dyadic resolution level by the principle of minimizing the Stein's Unbiased Risk Estimator for threshold estimates. It is smoothness adaptive, which means that if the unknown function contains abrupt changes or boundaries in the image, the reconstructed image also does.

IV.II BayesShrink

BayesShrink was proposed by Chang, Yu and Vetterli. The goal of this method is to minimize the Bayesian risk, and hence its name, BayesShrink. It uses soft thresholding and is sub-band-dependent, which means that thresholding is done at each band of resolution in the wavelet decomposition. Like the SureShrink procedure, it is smoothness adaptive. The Bayes threshold, t_B, is defined as:

$$t_B = \frac{\sigma^2}{\sigma_s}$$

Where $\sigma^2$is the noise variance and σ_s is the signal variance without noise. The noise variance σ^2 is estimated from the sub band HH1. From the definition of additive noise we have:

$$w(x,y) = s(x,y) + n(x,y)$$

Since the noise and the signal are independent of each other, it can be stated that

$$\sigma_w^2 = \sigma_s^2 + \sigma^2$$

σ_w^2 can be computed as shown below:

$$\sigma_w^2 = \frac{1}{n^2}\sum_{x,y=1}^{n} w^2(x,y)$$

The variance of the signal, $\sigma_s^2$is computed as:

$$\sigma_s = \sqrt{\max\,(\sigma_w^2 - \sigma^2, 2)}$$

V. CONCLUSION

In this review paper denoising techniques for the AWGN signal introduced in an image have been studied. This paper reviews the existing denoising algorithms, such as filtering approach; wavelet based approach. Different noise models including additive and multiplicative types are used. They include Gaussian noise, salt and pepper noise, speckle noise and Brownian noise. The filtering approach seems to be a better choice when the image is corrupted with salt and pepper noise. The wavelet based approach finds applications in denoising images corrupted with Gaussian noise. Selection of the denoising algorithm is application dependent. Hence, it is necessary to have knowledge about the noise present in the image so as to select the appropriate denoising algorithm.

REFERENCES

[1] Deepa Kundur, Student Member, and Dimitrios Hatzinakos,"A Novel Blind Deconvolution Scheme for Image Restoration Using Recursive Filtering ," IEEE Transactions On Signal Processing, Vol. 46, no. 2, February 1998.

[2] Xiumei Kang, Qingjin Peng, Gabriel Thomas and Yu, "Blind Image Restoration Using The Cepstrum Method" IEEE CCECE/CCGEI, Ottawa, May 2006.

[3] PunamPatil&R.B.Wagh "Implementation of Restoration of Blurred Image Using Blind Deconvolution Algorithm" IEEE 2013.

[4] Xiang Zhu, Member, IEEE, Scott Cohen, Member, IEEE, Stephen Schiller, Member, IEEE, and PeymanMilanfar, Fellow, IEEE Estimating Spatially Varying Defocus Blur From A Single Image, IEEE transactions on image processing, VOL. 22, NO. 12, DECEMBER 2013.

[5] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in Proc. IEEE CVPR, Aug. 2009, pp. 1964–1971.

[6] L. Xu and J. Jia, "Two-phase kernel estimation for robust motion deblurring," in Proc. ECCV, 2010, pp. 157–170.

[7] O.Whyte, J. Sivic, A. Zisserman, and J. Ponce, "Non-uniform deblurring for shaken images," in Proc. IEEE CVPR, Jun. 2010, pp. 491–498.

[8] A. Chakrabarti, T. Zickler, and W. T. Freeman, "Analyzing spatially varying blur," in Proc. IEEE CVPR, Jun. 2010, pp. 2512–2519.

[9] S. Bae and F. Durand, "Defocus magnification," Comput.Graph.Forum, vol. 26, no. 3, pp. 571–579, 2007.

[10] D. Zoran and Y. Weiss, "Scale invariance and noise in natural images," in Proc. IEEE 12th Int. Conf. Comput. Vis., Oct. 2009, pp. 2209–2216.

[11] Vikas Gupta, Rajesh Mahle, AshishShukla, " Removal of Gaussian Noise from Stationary Image using Shift Invariant Wavelet Transform" , 978-1-4673-5999-3/13, IEEE 213.